

# ACCELERATED BACHELOR OF SCIENCE (B.S.) AND MASTER OF SCIENCE (M.S.) IN COMPUTER SCIENCE

Students accepted into this selective program accomplish both the B.S. and M.S. degrees within five years by taking additional graduate courses within the first four years of the program. Up to two of these courses will count as open electives in the B.S. program and as didactic course work in the M.S. program.

## Program administration and minimum requirements

In order to be accepted students must:

1. Apply to the computer science graduate committee during the first semester after they have completed the last of the following sequence of classes:

Course	Title	Hours
CMSC 255	Introduction to Programming	4
CMSC 256	Data Structures and Object Oriented Programming	4
CMSC 257	Computer Systems	4
CMSC 302	Introduction to Discrete Structures	3
CMSC 303	Introduction to the Theory of Computation	3
CMSC 311	Computer Organization	3

2. Have a minimum GPA of 3.4 based on CMSC 255, CMSC 256, CMSC 257, CMSC 302, CMSC 303 and CMSC 311; all grades in any repeated courses will be included in computing this GPA. Students transferring these courses into the program will have the grades from their previous institution included in computing this GPA only for purposes of determining eligibility for this program.
3. Have a minimum overall GPA of 3.0.

In the last semester before graduating with the B.S. degree, the student will formally apply to the master's program. Providing the student has maintained a minimum GPA of 3.2 in the major, acceptance to the M.S. program is guaranteed. Accepted students are not required to complete the GRE for admission to the M.S. portion of the program.

## Special regulations in the accelerated program

Students accepted into the B.S.-M.S. program are allowed to transfer up to 12 graduate-level credits into the M.S. program, including up to six credit hours that were counted as open electives toward requirements for the B.S. degree.

After meeting all requirements for the B.S. degree, students in the program are eligible to take 600-level courses.

Apart from the exceptions above, all regulations outlined in the B.S. in Computer Science and M.S. in Computer Science bulletins apply toward the respective degrees.

## Typical program of study

Before graduating with the B.S. degree, students in the program are expected to:

- Take six graduate-level didactic credits that will count as open electives toward their B.S. degree (that is, toward the requirements on total number of credits, upper-level credits and toward graduation GPA, but not as the required three CMSC technical electives) and as didactic credits toward their M.S. degree.
- Take an additional six graduate-level didactic credit hours that will count toward their M.S. degree but not toward the B.S. degree. In particular, these cannot be used to satisfy the total and upper-level credit requirements in the B.S. degree nor in calculating the B.S. graduation GPA.

Students cannot count more than six credit hours of non-CMSC courses toward the M.S. degree. Any non-CMSC graduate credits require approval of the graduate committee.

The typical full program of study in the accelerated B.S.-M.S. program is as follows:

### Years 1-3

- Regular undergraduate program course work

### Year 3

- Application to the accelerated B.S.-M.S. program

### Year 4

- Remaining regular undergraduate program course work
- Six credit hours of CMSC 500-level courses, counted toward B.S. and M.S.
- Six credit hours of CMSC 500-level courses, counted toward M.S. only
- Application to the M.S. program

### Year 5

Regular graduate program course work: 18 credits of CMSC 500-level and 600-level courses, counted toward M.S. only.

### CMSC 101. Introduction to Computer Science. 3 Hours.

Semester course; 3 lecture hours. 3 credits. Prerequisite: MATH 141 or the equivalent with a minimum grade of C. An introduction to the work of computer scientists, including an overview of current research and application areas as well as career opportunities. Topics include problem-solving, the basics of computer organization, the software engineering life cycle, research resources and social and ethical aspects of technology. Additional topics also include binary, hexadecimal, two's complement, floating point representation, ASCII and Unicode.

### CMSC 191. Topics in Computer Science. 3 Hours.

Semester course; 3 lecture hours. 3 credits. May be repeated for credit. Prerequisite: permission of the instructor. This course will teach selected topics in computer science. See the Schedule of Classes for specific topics to be offered each semester and prerequisites.

**CMSC 245. Introduction to Programming Using C++. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: MATH 151 or satisfactory score on the Mathematical Placement Test. Students registering for CMSC 245 must have taken the VCU Mathematics Placement Test within the one-year period immediately preceding the beginning of the course. An exception to this policy is made in the case in which the stated alternative prerequisite course has been completed at VCU. Students are expected to have fundamental computer skills. Introduction to the concepts and practice of structured programming using C++. Problem-solving, top-down design of algorithms, objects, basic C++ syntax, control structures, functions and arrays. This course is intended for engineering majors.

**CMSC 246. Advanced Programming Using C++. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 245. Advanced programming in C++. Topics include program design, objects, classes, inheritance, files, strings, linked lists, stacks, queues, binary trees, recursion, and basic searching and sorting techniques. This course is intended for engineering majors.

**CMSC 255. Introduction to Programming. 4 Hours.**

Semester course; 3 lecture and 2 laboratory hours. 4 credits. Prerequisite: calculus-level placement on the VCU Mathematics Placement Test within the one-year period immediately preceding enrollment in the course, or MATH 151 or equivalent. Students are expected to have fundamental computer skills. Introduction to object-oriented programming using Java. Topics include problem-solving, top-down design of algorithms using control structures, methods, arrays, basic I/O, basic concepts of objects and classes in Java, Java classes for manipulating strings, and introduction to program testing, UML notation and integrated development environments. Students may not receive credit for both CMSC 255 and INFO 250.

**CMSC 256. Data Structures and Object Oriented Programming. 4 Hours.**

Semester course; 3 lecture and 2 laboratory hours. 4 credits. Prerequisite: CMSC 255 with a minimum grade of C; corequisite: CMSC 302. Advanced programming using Java. Topics include introduction to object-oriented design, inheritance, polymorphism, exceptions, interfaces, linked lists, stacks, queues, binary trees, recursion, and basic searching and sorting techniques. Continued focus on program testing and UML notation. Students may not receive credit for both CMSC 256 and INFO 350.

**CMSC 257. Computer Systems. 4 Hours.**

Semester course; 3 lecture and 2 laboratory hours. 4 credits. Prerequisite: CMSC 256 with a minimum grade of C. Topics include UNIX essentials; system programming in C; machine-level representation and organization of programs/data, arrays and pointers; types, structs and unions; strings; bit/byte operations; memory management; shell programming; input/output, including file handling; debugging; signals; network programming using sockets; program concurrency using forks and threads; experiments on program performance and optimization techniques.

**CMSC 302. Introduction to Discrete Structures. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 255 with minimum grade of C. Logic and proofs, sets, functions, sequences and sums, relations, graphs, trees, induction and recursion, advanced counting technique (recurrences).

**CMSC 303. Introduction to the Theory of Computation. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 302 or the equivalent with a grade of C or better. Complexity classes, grammars, automata, formal languages, Turing machines, computability.

**CMSC 311. Computer Organization. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 302 with minimum grade of C; corequisite: CMSC 257. Introduction to the basic organization of computers including elementary digital logic design, processor and arithmetic/logic unit design, data paths, memory hierarchy, I/O devices, instruction set architecture and addressing modes.

**CMSC 312. Introduction to Operating Systems. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 311 or EGRE 364. Computer systems design, I/O processing, secondary memory organization, command languages, memory management and job scheduling. Students will work in teams to design and implement an operating system simulation.

**CMSC 355. Software Engineering: Specification and Design. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 256 or EGRE 246, either with a minimum grade of C. Provides an overview of the software engineering process and software life-cycle models. Gives a detailed study of the analysis, specification and design phases. Students will work in teams to gain experience in software development methodology, developing specification and design documents and developing a prototype.

**CMSC 391. Topics in Computer Science. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. May be repeated for credit. Prerequisite: permission of the instructor. This course will teach selected topics in computer science. See the Schedule of Classes for specific topics to be offered each semester and prerequisites.

**CMSC 401. Algorithm Analysis with Advanced Data Structures. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 256 with a grade of C or better and CMSC 302 with a grade of C or better. Introduction to algorithm analysis and complexity classes. Advanced data structures topics including multiple linked lists, height-balanced trees, B-trees, hashing and graph representation; incorporating data structures into object-oriented design. Analysis of various searching and sorting algorithms. Algorithm design topics include divide-and-conquer, dynamic programming and greedy methods.

**CMSC 403. Programming Languages. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 256 and CMSC 303, both with a minimum grade of C. Survey of representative modern programming languages. Formal definition of programming languages including specifications of syntax and semantics. Precedence, infix, prefix and postfix notation. Global properties of algorithmic languages. Sub-routines, co-routines and tasks. List processing, string manipulation, data description and simulation languages. Run-time representation of program and data structures.

**CMSC 404. Compiler Construction. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 401 and 403. A first course in compiler theory and construction. Formal description of languages, underlying theory and design techniques for compilers, lexical analysis, syntax analysis, syntax-directed translation, intermediate languages, run-time system management, code generation, code optimization, compiler-building tools.

**CMSC 409. Artificial Intelligence. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 401 with a minimum grade of C and MATH 310. Covers problem spaces, problem-solving methods, game playing, knowledge representatives, expert systems, natural language understanding.

**CMSC 410. Introduction to Quantum Computing. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 401 and MATH 310, both with a minimum grade of B. Introduction to quantum information processing: state vectors and density operators, tensor product space, unitary evolution, no-go theorems, measurement, qubit, gate model of quantum computing, quantum complexity theory, quantum error correction, quantum algorithms, and quantum machine learning.

**CMSC 411. Computer Graphics. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 355 and MATH 310. Presents mathematical techniques for graphic development and transformation, curve and surface approximation and projections, graphical languages and data structures and their implementation, graphic modeling.

**CMSC 412. Social Network Analysis and Cybersecurity Risks. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 401 with a minimum grade of C. Covers network models, link prediction and analysis, centrality measures, random networks, power-laws and preferential attachment, small world phenomenon and decentralized search, community structure, information propagation in networks, and security and privacy issues in OSNs.

**CMSC 413. Introduction to Cybersecurity. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 401 with a minimum grade of C. This course provides introduction and basic concepts of computer security, cyber attacks, cyber defense, cyber forensics and cyber ethics.

**CMSC 414. Computer and Network Security. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 401 with a minimum grade of C. Corequisite: CMSC 312. This course covers the best practices of computer systems and network security. Key topics include security architecture, cryptographic systems and security management tools.

**CMSC 415. Introduction to Cryptography. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 401 with a minimum grade of C. This course provides a rigorous and theoretical introduction to modern cryptography. Key topics include symmetric key encryption and authentication, public key encryption, and digital signatures.

**CMSC 416. Introduction to Natural Language Processing. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 401 with a minimum grade of C. Covers rule-based and statistical methods for creating computer programs that analyze, generate and understand human language. Topics include regular expressions and automata, context-free grammars, probabilistic classifiers, and machine learning. Word-level, syntactic and semantic processing are all considered. Application to real-world problems such as spell-checking, Web search, automatic question answering, authorship identification and developing conversational interfaces.

**CMSC 420. Software Engineering: Project Management. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 355 with a minimum grade of C. Study of the logistics of team software development. Students work in teams to gain experience in software management and develop the components of a larger software product. Topics include risk management, project planning, quality management, configuration management and software testing.

**CMSC 428. Mobile Programming: iOS. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 355, with a minimum grade of C. This course covers the fundamentals of Swift, Xcode and iOS for programming and design of iOS applications. Background in object-oriented programming and access to a computer with Xcode platform is required.

**CMSC 435. Introduction to Data Science. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 401 with a minimum grade of C. This course covers understanding, representation, storage, retrieval, preprocessing and analysis of data. Specific topics include data quality and preprocessing, database management systems, data warehouses, selected methods for scalable unsupervised and supervised data analysis, and assessment of results generated by these methods. Students will be engaged in analysis of real-life data from data preprocessing, through data analysis, to the assessment of a knowledge product.

**CMSC 451. Senior Project I. 3 Hours.**

Semester course; 3 laboratory hours. 3 credits. Prerequisites: CMSC 355 with minimum grade of C; and UNIV 200 or HONR 200 or equivalent. Enrollment is restricted to computer science majors with senior standing who have 24 credits in computer science courses. Capstone project or experience for the computer science major; research and presentation methods in computer science. Each student will participate, either individually or as part of a team, in a project or other experience approved by the course coordinator or sponsored by another computer science faculty member. Each student will write and revise a research paper on a technical topic associated with his or her project or experience. Students will submit a detailed written description of their proposed project or experience and will present orally some aspect of what they have learned and/or done during the semester. (This course cannot be counted as upper-level CMSC electives for students graduating under bulletins prior to 2008-09.)

**CMSC 452. Senior Project II. 3 Hours.**

Semester course; 1 lecture and 2 laboratory hours. 3 credits. Prerequisites: CMSC 451 and CMSC 508, both with a minimum grade of C. Enrollment is restricted to students with senior standing in the computer science department. Capstone project or experience for the computer science major; research and presentation methods in computer science; ethical, legal and social issues in computing; and professional responsibilities of computer scientists. Each student will participate, either individually or as part of a team, in a project or other experience approved by the course coordinator or sponsored by another computer science faculty member. Each student will write and revise a research paper on a technical topic associated with his or her project or experience. Students must continue on the same project that was started in CMSC 451. A final project report and presentation, which will include a discussion of associated legal, social and/or ethical issues, are due at the conclusion of the two-semester project or experience. (This course cannot be counted as upper-level CMSC electives for students graduating under bulletins prior to 2008-09.)

**CMSC 491. Topics in Computer Science. 1-3 Hours.**

Semester course; variable hours. 1-3 credits. May be repeated for credit with different content. Prerequisite: permission of instructor. This course will cover selected topics in computer science. See the Schedule of Classes for specific topics to be offered each semester.

**CMSC 492. Independent Study. 2-4 Hours.**

Semester course; variable hours. 2, 3 or 4 credits per semester. Maximum 4 credits per semester; maximum total of 6 credits. Generally open only to students of junior or senior standing who have acquired at least 12 credits in the departmental discipline. Determination of the amount of credit and permission of instructor and department chair must be procured prior to registration of the course. The student must submit a proposal for investigating some area or problem not contained in the regular curriculum. The results of the student's study will be presented in a report.

**CMSC 493. Computer Science Internship. 1-3 Hours.**

Semester course; variable hours. 1-3 credits. May be repeated for up to 3 credits. Prerequisites: CMSC 401 and CMSC 403. Approval of Computer Science Undergraduate Credentials Committee is required prior to registration. A minimum of 30 clock hours per credit in an information technology environment. The internship is designed to provide practical experience in the computing industry. Student must present a written report reflecting upon internship experience. Graded as pass/fail. Not applicable toward the computer science major.

**CMSC 501. Advanced Algorithms. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 401 or equivalent; graduate standing or acceptance into accelerated B.S. to M.S. program in computer science. Advanced graph algorithms, advanced data structures, applied numerical algorithms, optimization methods, approximation methods for hard graph and string problems, and computational geometry algorithms.

**CMSC 502. Parallel Algorithms. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 312 and CMSC 401, graduate student standing or acceptance into the five-year accelerated B.S. and M.S. program in computer science. Software and hardware mechanisms for providing mutual exclusion in uniprocessor and multiprocessor environments. Architectural issues including pipeline design, superscalar computers, multiprocessors, memory systems, peripherals, interfacing techniques, networks, performance and software issues. Design and uses of parallel algorithms to solve concurrency problems in a distributed environment including message passing and remote procedure calls. Students will work in teams (as well as on individual projects) to design and implement parallel algorithms.

**CMSC 506. Computer Networks and Communications. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 312. Theoretical and applied analysis of basic data communication systems; design of networks in the framework of the OSI reference model; Local and Wide Area Networks; performance analysis of networks; error control and security. Students will work in teams to design and implement a small computer network. Crosslisted as: EGRE 526.

**CMSC 508. Database Theory. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 303 with a minimum grade of C. Design and implementation of relational database systems. Emphasis is placed on entity-relationship diagrams, relational algebra, normal forms and normalization. Introduction to SQL. Discussion of physical level issues. Students will be required to complete a design project and give an oral presentation of the project. Not applicable toward the M.S. in Computer Science or the Ph.D. in Engineering, computer science concentration.

**CMSC 510. Regularization Methods for Machine Learning. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Enrollment restricted to students with graduate standing in computer science or related discipline such as bioinformatics or acceptance into five-year accelerated program in computer science. The course will assume undergraduate-level background in algorithms, linear algebra, calculus, statistics and probability. Upon successful completion of this course, the student will be able to understand recent advances in machine learning and apply machine-learning tools that go beyond learning from data, as well as have the ability to incorporate additional knowledge about the learning problem. Topics covered will include optimization-based view of supervised machine learning; classical regularization approaches including weight decay and Lasso; regularization terms incorporating additional knowledge about structures in the feature space, including group lasso and graph-based regularization terms; semi-supervised learning using graphs linking unlabeled and labeled samples.

**CMSC 512. Advanced Social Network Analysis and Security. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Pre- or corequisites: CMSC 412 and CMSC 501. Enrollment requires graduate student standing in computer science or related discipline such as bioinformatics, or acceptance into five-year accelerated program in computer science. The purpose of the course is to teach algorithms for analyzing social networks and complex systems. The focus will be on understanding the inner workings of algorithms using in-network analysis and security threats in online social network sites. Topic covered will include modeling social and technological networks, methods for analyzing structure and dynamical processes on the network, and security and privacy issues in online social networks such as inference attacks, network anonymization, sybil attacks and defense, social bots.

**CMSC 516. Advanced Natural Language Processing. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Enrollment restricted to graduate students in computer science or related discipline, or acceptance into five-year accelerated program in computer science. Upon successful completion of this course, the student will be able to understand recent advances in natural language processing and apply NLP algorithms and techniques for processing unstructured text. Word-level, syntactic and semantic processing will be considered. Specific topics include rule-based and statistical methods for creating computer programs that analyze, generate and understand human language. Regular expressions and automata, context-free grammars, probabilistic classifiers and machine learning. Applications to real-world problems such as spell-checking, Web search, automatic question answering, authorship identification and developing conversational interfaces.

**CMSC 525. Introduction to Software Analysis, Testing and Verification. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 401 and 403, graduate student standing or acceptance into the five-year accelerated B.S. and M.S. program in computer science. An introduction to concepts and techniques used in the analysis of software for certain properties. Using analytic results to derive test data and verify the correct implementation of programs. Flow graphs, fault/failure model, theoretical and practical limitations. Control flow, data flow and error flow analyses. Testing strategies including random, structural, mutation and error flow. Software metrics.



**CMSC 526. Theory of Programming Languages. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 403, graduate student standing or acceptance into the five-year accelerated B.S. and M.S. program in computer science. An introduction to the formal semantics of programming languages, logic programming and functional programming. Topics include denotational semantics, attribute grammars, Backus Formal Functional Programming, fixed point semantics, model-theoretic semantics and PROLOG.

**CMSC 591. Topics in Computer Science. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. May be repeated for credit. Prerequisites may vary. Permission of the instructor required. Course is open to graduate students and students accepted into the five-year accelerated B.S. and M.S. program in computer science. A study of selected topic(s) in computer science at the graduate level. See the Schedule of Classes for specific topics to be offered each semester.

**CMSC 601. Convex Optimization. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 501 or permission of instructor. Enrollment restricted to students with graduate standing in computer science or related discipline. A background in undergraduate-level linear algebra is assumed. Convex sets and functions. Convex optimization problems: Linear, quadratic, semi-definite and cone programs. Duality theory. Approximation algorithms for NP-complete integer optimization problems via semi-definite relaxations and rounding schemes. Algorithms for optimization, such as gradient descent, proximal descent, alternating directions method of multipliers, interior point methods.

**CMSC 602. Operating Systems. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 502. A study of operating systems including those in multiprocessor and distributed environments. I/O programming, resource management (including processor and memory management), security and system performance evaluation.

**CMSC 603. High Performance Distributed Systems. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Enrollment restricted to students with graduate standing in computer science or related discipline or acceptance into five-year accelerated program in computer science. The course will assume undergraduate-level background in algorithms, data structures and parallel programming. Upon successful completion of this course, the student will be able to understand the concepts underlying distributed systems; analyze problems to identify performance bottlenecks, parallelization opportunities and concurrency issues in a distributed environment; create distributed and scalable implementations using multiple hosts/GPUs; design and implement algorithms using Hadoop, Spark and CUDA.

**CMSC 605. Advanced Computer Architecture. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: EGRE 426 or with permission of instructor. This course will focus on the design and analysis of high performance computer architectures. Topics investigated include pipeline design, superscalar computers, multiprocessors, memory systems, peripherals, interfacing techniques, networks, performance and software issues. Crosslisted as: EGRE 635.

**CMSC 608. Advanced Database. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 508. Topics discussed include: handling of missing information; the relationship between relational calculus, relational algebra and SQL; logic databases; distributed databases; outer joins; and transaction processing. Emphasis is placed on theoretical issues involved in these topics. In addition students will work in teams to develop a working database application.

**CMSC 610. Algorithmic Foundations of Bioinformatics. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: Graduate student standing or acceptance into five-year accelerated program in computer science or related discipline such as bioinformatics. The purpose of the course is to teach algorithms for analyzing biological and medical data. The focus will be on understanding the inner workings of algorithms used in bioinformatics tools. Topic covered will include algorithms for assembling and searching biological sequences, finding patterns associated with disease, and exploring biological networks.

**CMSC 611. Computer Multimedia. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: permission of instructor. Study of computer multimedia techniques relating to images, sound, video and text. Emphasis on compression techniques and standard storage formats. This course is programming-intensive.

**CMSC 612. Game Theory and Security. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 401, graduate student standing or acceptance into the five-year accelerated B.S. and M.S. program in computer science. The course will provide an introduction to game theory and mechanism design concepts. Lectures cover topics such as introduction of games, equilibrium concepts, computation of game-theoretic solution concepts, mechanism, and issues in game theory and mechanism design.

**CMSC 615. Cryptocurrency and Blockchain Techniques. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Enrollment is restricted to students with graduate standing in computer science or a related discipline. The course will assume undergraduate-level background in algorithms, data structures and programming. Upon successful completion of this course, the student will be able to understand the major concepts about cryptocurrency and blockchain techniques; be familiar with major blockchain applications as well as real-world issues; understand the underlying consensus mechanisms in the Bitcoin system and other alternative cryptocurrency systems; analyze the security of Nakamoto consensus; understand, write and execute smart contracts using an Ethereum-like platform.

**CMSC 618. Database and Application Security. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Theory and practice of database and software security focusing in particular on some common database software security risks and on the identification of potential threats and vulnerabilities. Crosslisted as: CISS 618.

**CMSC 619. The Design and Specifications of User Interfaces. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: Graduate standing and permission of instructor. Requires knowledge of first order predicate calculus and context-free languages. Focuses on human-computer interface design principles and methodology and formal specifications of user interfaces.

**CMSC 620. Applied Cryptography. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Provides a comprehensive survey of modern cryptography. Included are techniques of enciphering and deciphering messages using cryptographic algorithms, block ciphers and block cipher modes, hash functions and message authentication codes, public key cryptography and digital signatures, and steganography. Crosslisted as: CISS 624.

**CMSC 621. Theory of Computation. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: graduate student standing and permission of instructor. Discussion of the complexity and computability of problems and programs. Topics will include unsolvability, universal programs and abstract complexity.

**CMSC 622. Network and Operating Systems Security. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CISS 624. Studies the principles of network security and secure operating systems. Included are topics relating to the use of intrusion detection, intrusion prevention and other related tools. Crosslisted as: CISS 622.

**CMSC 623. Cloud Computing. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 622. Provides an introduction to cloud computing architecture and cloud computing security. The course covers the basic concepts of cloud computing, including memory virtualization, device virtualization and related security problems in cloud computing.

**CMSC 624. Software Quality Assurance. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisites: a course in software engineering and graduate standing in computer science, or permission of instructor. A study of issues that affect the quality of software and of methodology to assure that software products are of the desired quality. This also includes issues in assessing product quality as well as the process by which the software is produced. Topics include various methodologies, standards, metrics and tools.

**CMSC 625. Advanced Software Analysis, Testing and Verification. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 525. Studies the concepts and techniques used in the analysis of software and the derivation of test data. Focuses on software metrics and reliability; construction of tools to aid software analysis and testing. Requires students to review seminal and current papers from the literature, and lead their discussion in class.

**CMSC 628. Mobile Networks: Applications, Modeling and Analysis. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Enrollment is restricted to students with graduate standing in computer science or related discipline. The course will assume undergraduate-level background in algorithms, data structures, programming and networks. Upon successful completion of this course, the student will be able to understand the major concepts about mobile networks such as device-to-device communication technologies, mobility models and coverage; be familiar with various mobile network types (e.g., mobile social networks, delay tolerant networks, overlay networks, vehicular networks and cellular networks) and devices (e.g., smartphones, femtocells, WiFi), learn how to model mobile networks with stochastic processes and real datasets; be able to use different networking simulators; and understand various routing algorithms and analyze their behavior.

**CMSC 630. Image Analysis. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Enrollment restricted to students with graduate standing in engineering or science or by permission of instructor. Introduces theoretical and practical aspects of computer vision for image processing and understanding. It provides a comprehensive walkthrough from basics of image preparation to using computational intelligence tools for knowledge discovery from images. The course will cover basics of image processing and computer vision, including image sampling and quantization, color, pixel-based operations, image filtering, morphological image processing, and image transforms; information extraction including segmentation and feature extraction; pattern recognition for computer vision: classification, novelty and object detection, image understanding, learning from video streams, and tensor-based methods. Examples will include medical image analysis, object recognition in ground and aerial photographs and hyperspectral imaging.

**CMSC 635. Knowledge Discovery and Data Mining. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 401 or corequisite: CMSC 501. Enrollment restricted to students with graduate standing in computer science or related discipline such as bioinformatics or acceptance into five-year accelerated program in computer science. Covers knowledge discovery and data mining concepts, tools and methods; provides hands-on experience based on a project involving analysis of large real-life data. Topics include the knowledge discovery process, data storage and representation, preprocessing algorithms for missing data imputation, feature selection and discretization; unsupervised learning algorithms for clustering and association mining; supervised learning algorithms including decision trees, Bayesian models and introduction to support vector machines and neural networks; ensemble learning; protocols and measures for validation of predictive models; and data security and privacy issues.

**CMSC 636. Artificial Neural Networks and Deep Learning. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Enrollment restricted to students with graduate standing in computer science. The course will assume undergraduate-level background in programming, algorithms, linear algebra, calculus, statistics and probability. Topics ranging from fundamental learning rules, functional, cascade correlational, recurrent and gradient descent networks, to neocognitron, softmax, deep convolutional networks, autoencoders and pretrained deep learning (restricted Boltzmann machines). Students will be required to work in teams on a class paper.

**CMSC 678. Statistical Learning and Fuzzy Logic Algorithms. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: MATH/STAT 309 or 310. The course considers two central problems in modern science and engineering: i) the problem of statistical learning from examples (empirical data) and ii) the problem of embedding existing human knowledge into workable mathematics. Topics include: examples of multivariate functional mapping, basics of classic classification and regression, support vector machines as a learning paradigm based on structural risk minimization, fuzzy logic algorithms, basics of multi-class classification over high dimensional spaces, curve and surface fittings, multivariate function approximation and nonlinear optimization; fuzzy logic systems; crisp and fuzzy sets, linguistic variables, fuzzy set theory; if-then rules, fuzzy interference, fuzzification and defuzzification, neuro-fuzzy paradigms.

**CMSC 691. Special Topics in Computer Science. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. May be repeated for credit. Prerequisites: at least one graduate-level computer science course pertaining to the topic area and permission of instructor. An advanced study of selected topic(s) in computer science at the graduate level. See the Schedule of Classes for specific topics to be offered each semester.

**CMSC 692. Independent Study. 1-3 Hours.**

Semester course; 1-3 variable hours (to be arranged). 1-3 credits. Enrollment restricted to students with graduate standing and consent of instructor. Independent study done under the supervision of a faculty member. The student must identify a faculty member willing to supervise the research and submit a proposal for approval by the computer science graduate committee no later than the 10th week of the prior semester. A written report and an oral presentation are required upon completion of the research project. Graded as Pass/Fail.

**CMSC 697. Directed Research. 1-15 Hours.**

Semester course; variable hours (to be arranged). 1-15 credits. May be repeated for credit. A total of 3 credits may be used to fulfill the M.S. in Computer Science thesis requirement. Prerequisite: graduate standing. Independent research culminating in the writing of the required thesis or dissertation. The student must identify a faculty member willing to supervise the research and submit a proposal to the computer science graduate committee no later than the 10th week of the prior semester. This proposal must be approved before the student can register for the course. Graded as S/U/F.

**CMSC 701. Research Methods. 3 Hours.**

Semester course; 3 lecture hours. 3 credits. Prerequisite: Ph.D. standing or permission of instructor. Covers the principles of conducting a research project, reporting the findings in the form of a journal paper and promoting the research through public presentations. Students learn to write grant proposals and practice reviewing research papers and grant proposals. The main emphasis of the course is writing a paper and a grant proposal in a format compliant with NSF, NIH or DoD guidelines.

**CMSC 702. Computer Science Seminar. 1 Hour.**

Semester course; 1 seminar hour. 1 credit. May be repeated for credit. Enrollment restricted to students in the doctoral program in computer science. Students will attend a weekly research seminar in which the topic and speaker will change each week in order to cover a broad range of subjects at the forefront of computer science research. Students will have to present and to write a report on at least one seminar presented by other speakers. The objective is to expose students to research topics and scholars in the field of computer science as well as to provide them experience in delivering and critiquing seminar talks. Graded as satisfactory/unsatisfactory.